waveguide **3144** through its bottom surface. This occurs for each pointer **3011** as it contacts the touch surface at a respective touch point.

[0122] As each touch point is moved along the display surface **3015** of the touch panel **3014**, the compression of the resilient diffusion layer **3146** against the optical waveguide **3144** occurs and thus escaping of IR light tracks the touch point movement. During touch point movement or upon removal of the touch point, decompression of the diffusion layer **3146** where the touch point had previously been due to the resilience of the diffusion layer **3146**, causes escape of IR light from optical waveguide **3144** to once again cease. As such, IR light escapes from the optical waveguide **3144** only at touch point location(s) allowing the IR light to be captured in image frames acquired by the imaging device.

[0123] The imaging device **3032** captures two-dimensional, IR video images of the third mirror **3030**. IR light having been filtered from the images projected by projector **3022**, in combination with the cabinet **3016** substantially keeping out ambient light, ensures that the background of the images captured by imaging device **3032** is substantially black. When the display surface **3015** of the touch panel **3014** is contacted by one or more pointers as described above, the images captured by IR camera **3032** comprise one or more bright points corresponding to respective touch points. The processing structure **3020** receives the captured images and performs image processing to detect the coordinates and characteristics of the one or more touch points based on the one or more bright points in the captured images. The detected coordinates are then mapped to display coordinates and interpreted as ink or mouse events by the processing structure **3020** for manipulating the displayed image.

[0124] The host application tracks each touch point based on the received touch point data, and handles continuity processing between image frames. More particularly, the host application receives touch point data from frames and based on the touch point data determines whether to register a new touch point, modify an existing touch point, or cancel/delete an existing touch point. Thus, the host application registers a Contact Down event representing a new touch point when it receives touch point data that is not related to an existing touch point, and accords the new touch point a unique identifier. Touch point data may be considered unrelated to an existing touch point if it characterizes a touch point that is a threshold distance away from an existing touch point, for example. The host application registers a Contact Move event representing movement of the touch point when it receives touch point data that is related to an existing pointer, for example by being within a threshold distance of, or overlapping an existing touch point, but having a different focal point. The host application registers a Contact Up event representing removal of the touch point from the display surface **3015** of the touch panel **3014** when touch point data that can be associated with an existing touch point ceases to be received from subsequent images. The Contact Down. Contact Move and Contact Up events are passed to respective elements of the user interface such as the graphic widgets, or the background, based on the element with which the touch point is currently associated, and/or the touch points current position. Routines similar to those previously described can be run by the host application allowing the host application to detect multi-pointer gestures.

[0125] Another embodiment of the thresholding process **2400** suitable for use in the FTIR interactive input system described above is shown in FIG. **29**. In the FTIR thresholding process **2500**, the current image **2502** and the background image **2504** are compared in a similarity calculation **2506** to identify pointer contacts. At step **2508** a primary threshold is applied to the touch points. This threshold can be user adjustable as described in thresholding process **2400** to assign certain threshold properties to certain sizes of pointers.

[0126] At step **2510**, the image is segmented so that only those regions where pointer contacts appear are processed by the system to reduce the processing load. In step **2512**, the average brightness and standard deviation of pixels inside each segmented region are calculated, and in step **2514**, the threshold of each pointer is set to the average brightness value plus a multiple of the standard deviation in brightness. This threshold is typically set at about 1 standard deviation from the mean. The pointer contact is now represented by a virtual pointer having the threshold size.

[0127] In the FTIR system described above, it is possible to estimate the contact pressure applied by a user when touching the touch area by measuring the changing pointer size or brightness. Process **2600** in FIG. **26** outlines the pointer contact pressure estimation system. As with process **2500**, the current image **2602** and the background image **2604** are compared in the similarity calculation **2606** to identify pointer contacts. At step **2608** a primary threshold is applied to the touch points. This threshold can be user adjustable as described in thresholding process **2400** to assign certain threshold properties to certain sizes of pointers.

[0128] At step **2610**, the image is segmented so that only those regions where pointer contacts appear are processed by the system to reduce the processing load. In step **2512**, the average brightness and standard deviation of pixels inside each segmented region are calculated. At step **2620**, the pressure is estimated using the using the pointer contact brightness calculated in step **2612** and normalized using the upper and lower background levels. The upper background level **2616** is then updated with feedback from the calculated pressure.

[0129] At step **2614**, a background analysis is performed by averaging the brightness and standard deviation of the background image **2604**. At step **2618**, the lower background level is set to the average background brightness level minus one standard deviation. At step **2616**, the upper background level is set to an arbitrary reasonable default value.

[0130] The background image **2504** is continuously updated by blending some areas of the current image devoid of pointers on a frame by frame basis. When pointers dwell beyond a certain threshold time, they are ignored by the pointer recognition software as inactive pointers, such as a hand, mouse, cup, etc. resting on the input surface. When the latent pointer is removed, the background image is updated immediately to allow contact detect in that region.

[0131] As mentioned above, the gestures described herein are merely examples of gestures that may be used with the interactive input system. As one of skill in the art will appreciate, other whole hand or multiple touch point gestures that may be used in application associated with such interactive input system can employ similar routines as outlined above.

[0132] Rather than employing an illuminated bezel, the assembly may comprise a reflective or retroreflective bezel that reflects radiation emitted by radiation sources associated with the imaging devices so that the imaging devices see white bands in the absence of pointers.